1

The Nature of the Beast

In 1962, THE FIRST MODERN VIDEO game was invented by some hackers at MIT. It was called *Spacewar!*, and it ran on a DEC PDP-1, the world's first mini-computer, connected to a CRT display. One of the game's designers explained that the game was born as a group sat around trying to figure out what sort of "interesting displays" they could create for the CRT with some pattern-generating software they had developed. "We decided that probably, you could make a two-dimensional maneuvering sort of thing, and decided that naturally the obvious thing to do was spaceships." The MIT hackers weren't the only ones to invent *Spacewar!*. As Alan Kay noted, "the game of *Spacewar!* blossoms spontaneously wherever there is a graphics display connected to a computer" (Brand 1974).

Why was *Spacewar!* the "natural" thing to build with this new technology? Why not build a pie chart, an automated kaleidoscope, or a desktop? Its designers identified *action* as the

In 1962, I was given my first computer. It was a prize for a Halloween costume contest run by the local hardware store. The manager handed me what looked like a grey plastic box, with the word "Eniac" embossed on it. "It's a computer," he explained. He demonstrated its operation. First he showed me a card with this question printed on it: "What is the distance of the earth from the sun?" He inserted the card into the plastic box and turned a crank. A card was ejected from the other side of the device that showed the number 92,876,479.56. I was amazed. "You see?" he said excitedly. "It can answer questions. All kinds of questions. Here are the questions, right here." He brandished a packet that presumably contained all of the important questions one might ever want to ask. "All you have to do is feed them into the computer." I took the card from him and turned it over to discover that the question was printed on the back. The plastic Eniac had simply flipped the card over.

key ingredient, and conceived *Spacewar!* as a game that could provide a good balance between thinking and doing for its players. They regarded the computer as a machine naturally suited for representing things that you could see, control, and play with. Its interesting potential lay not simply in its ability to perform calculations, but in its capacity to co-create and represent actions with human participants.

The Interface

Why don't we look at everything computers do in the way that the *Spacewar!* hackers did? Consider the following question: What is being represented by a human-computer interface?

1. A way for a person to communicate with a computer.
2. A way for a computer to communicate with a person.
3. A surface through which humans and computers can communicate.
4. A way for humans and computers to construct actions together.

Number three comes close, but it implies a membrane or separation between the human and the computer. But the object is not the membrane; rather it is the action co-created by the human and technical forces at play. The difference in emphasis may be the impetus of the trend toward replacing the term "human-computer interface" with "human-computer interaction" in recent years.

There are two major reasons for belaboring such a seemingly obvious point. First, it wasn't always true—and the design disciplines for applications and interfaces still bear the marks of that former time. Second, reconceptualizing what computers do as enabling and representing actions that involve both human and technological participants suggests a design philosophy that diverges significantly from much of the received wisdom about interface design.

Provenance of the Interface

The notion of the "human-computer interface" was presaged by the field of human factors engineering, or human factors design. This discipline was born with the design of airplanes during WWII and the famous Link Trainer simulations that helped pilots safely learn how to fly by instruments. The field was informed by earlier work, including the famed "time motion studies" conducted by Frank and Lilian Gabreth in the 1920s. In fact, there is evidence that the closely related field of ergonomics was a design concern even in ancient Greece (see Marmaras et al. 1999). Human factors and ergonomics are concerned with taking the human's physical and cognitive abilities into account in the design of things humans use, as the sidebar illustrates with the evolution of the automobile "interface." An important characteristic of the human factors world before computerization was that the elements of the "interface"—the chair or airplane—were fixed in space and existed with fixed operational characteristics. The plasticity of the human-computer interface created huge new problems and opportunities for the human factors field. The "interface" was a powerful bridge, and design began to rely more upon cognitive aids such as metaphors rather than upon the characteristics of the body *per se*.

"Interface" became a trendy (and lucrative) concept in the 1980s and 1990s—a phenomenon that is largely attributable to the introduction of the Apple Macintosh. Interface design was concerned with making computer systems and applications easy to use (or at least usable) by humans. When we thought of human-computer interfaces in those days, we were likely to visualize icons and menu bars or perhaps command lines and blinking cursors. But, of course, many conceptions came before as well as after.

John Walker (founder and president of Autodesk, Inc.) provides an illuminating account of the "generations" of user interface design (Walker 1990). In the beginning, says Walker, there was a one-on-one relationship between a person and a computer through the knobs and dials on the front of massive early machines like the ENIAC. The advent of punched cards and batch processing replaced this direct human-computer interaction with a transaction mediated by a computer operator. Time-sharing and the use of "glass teletypes" reintroduced direct human-computer interaction and led to the command-line and menu-oriented interfaces with which the senior citizens of computing (people over forty) are probably familiar. Walker attributes the notion of "conversationality" in human-computer interfaces to this kind of interaction, in which a person does something and a computer responds—a tit-for-tat interaction.

This simplistic notion of conversation led many early interface specialists to develop a model of interaction that treats human and computer as two distinct parties whose "conversation" is mediated by the screen. But as advances in linguistics demonstrated, there is more to conversation than tit for tat. Dialogue is not just linearized turn-taking in which I say something, you go think about it, then you say something, I go think about it, and so

Automobile Interfaces

Interface for the Pontiac Firebird III.
Source: www.oldcarmanualproject.com

Interface design often is initially fluidly variant and later rather sticky, or conservative, as is seen in the history of automobiles. Automotive control interfaces initially resembled small boats in that they were steered with tillers (often with the driver in the rear) and featured hand controls for power and braking. A profusion of such designs existed in the 19th century. These evolved at the turn of the 20th century into a canonical set of controls that we would easily recognize today, with the steering wheel placed in front of an off-side driver in the front. The first car with a steering wheel was the single seat 4hp Panard racer in 1894, and the first production automobile with the driver and steering wheel on the left was Thomas B. Jeffery's 1904 Rambler (the 1903 Rambler was steered with a tiller).

The General Motors turbine-powered Firebird III concept car of 1957 was a notable exception inspired by the jet fighters of the time; the driver "piloted" it with a control stick placed between the car's two seats. This uni-control moved forward for acceleration, backwards for braking, side-to-side for steering, and twisted for shifting gears. It was located on the center arm rest between the two seats (each with an individual bubble canopy). This did not catch on as a control design. The seven tail fins, pop-out air-drag brakes, and titanium body were kinda cool, though.

It is likely that self-driving cars will continue to have vestigial steering wheels and brake/accelerator pedals well after the Robot Revolution, for reasons having to do with the stories we tell ourselves about power and control. These will be on the left side, except in Britain and Australia. But our robot masters will seldom let us drive unconstrained.

—Rob Tow, Ex-Xerox PARC Scientist and Part-time Abalone Diver

on. An alternative model of conversation employs the notion of common ground, as described by Herbert H. Clark and Susan E. Brennan (1990):

> It takes two people working together to play a duet, shake hands, play chess, waltz, teach, or make love. To succeed, the two of them have to coordinate both the content and process of what they are doing. Alan and Barbara, on the piano, must come to play the same Mozart duet. This is

> coordination of content. They must also synchronize their entrances and exits, coordinate how loud to play forte and pianissimo, and otherwise adjust to each other's tempo and dynamics. This is coordination of process. They cannot even begin to coordinate on content without assuming a vast amount of shared information or common ground—that is, mutual knowledge, mutual beliefs, and mutual assumptions (see Clark and Carlson 1982, Clark and Marshall 1981, Lewis 1969, Schelling 1960). And to coordinate on process, they need to update, or revise, their common ground moment by moment. All collective actions are built on common ground and its accumulation.

In her work in applying the notion of common ground to human-computer interfaces, Brennan (1990a) suggests that common ground is a jointly inhabited "space" in which meaning takes shape through the collaboration and successive approximations of the participants. Brennan's work was aimed at designing human-computer interfaces so that they offer means for establishing common ground ("grounding") that are similar to those that people use in human-to-human conversation, such as interruptions, questions, and utterances and gestures that indicate whether something is being understood (Brennan 1990b).

Successful graphical interfaces, exemplified early on by the Macintosh, explicitly represented part of what Clark called the "perceptual common ground" of interaction through the appearance and behavior of objects on the screen (Clark 1996). Some of what goes on in the representation is exclusively attributable to either the person or the computer, and some of what happens is a virtuous artifact of a collaboration in which the traits, goals, and behaviors of both are inseparably intertwined.

The concept of common ground not only provides a superior model of the conversational process, but it also supports the idea that an interface is not simply the means whereby a person and a computer represent themselves to one another; rather, it forms a shared context for action in which both are agents.[1] When the old tit-for-tat paradigm intrudes, the "conversation" is likely to break down, once again relegating person and computer to opposite sides of a "mystic gulf"[2] filled with hidden processes, arbitrary

1. This book employs the noun "agent" to mean *one who initiates action*. This definition is consistent with Aristotle's use of the concept in the *Poetics*.

2. The term "mystic gulf" is attributed to composer Richard Wagner to refer to the gap between audience and actors created by the orchestra pit.

understandings and misunderstandings, and power relationships that are competitive rather than cooperative. Mistakes, unanticipated outcomes, and error messages are typical evidence of such a breakdown in communication, in which the common ground becomes a sea of misunderstanding.

Interface Metaphors

The notion of interface metaphors was introduced to provide a conceptual scheme for people that would guard against such misunderstandings by deploying familiar objects and environments as stakes in the common ground: the anchoring expectations. The most famous of these is the desktop metaphor, first developed by Alan Kay at Xerox PARC in 1970, borrowing from some of the work of Douglas Engelbart at the Stanford Research Institute (SRI) in the 1960s. The Xerox Alto (1973) was the first computer that used the desktop metaphor and a graphical user interface (GUI), followed in 1981 by the Xerox Star workstation and the Apple Lisa in 1983. The first broad exposure of the desktop metaphor was in 1984 with the introduction of the Apple Macintosh, intended as a computer for the general public rather than for business use. It employed graphical icons to represent individual files as "documents" and hierarchical organizational units as "folders." Rumors of the death of the desktop have been highly exaggerated. Although it has sprouted many non-desktop affordances over the years (e.g., scroll bars and docks), we can still see its fundamental outlines in contemporary personal computers as well as in the arrangement of icons and what they mean on smartphones.

But even "good" metaphors don't always work as intended. Several years after the introduction of the Mac, in an informal survey of Macintosh-literate university students, for instance, many people failed to employ the word "desktop" anywhere in their description of the Finder.[3] Where an interface metaphor diverges significantly from its real-world referent, people proceed by accounting for the behaviors of particular "objects" on the screen with *ad hoc* explanations of system operation that are often incorrect: a "naïve physics" of computing (see Owen 1986). In such cases, metaphors do not serve as "stakes in the common ground," but rather as cognitive

3. The Macintosh Finder is an application for managing people's file systems and for launching other applications. It comes with the system and is automatically launched when the machine is turned on.

mediators whose labels may be somewhat less arcane (but possibly more ambiguous) than a computer scientist's jargon.

Since the introduction of the Mac, we have seen a variety of interface metaphors, both local and global. With the advent of the World Wide Web, we began to speak of a Web page as if it were a page in an enormous book. When things turned out to be a little more complicated, with hyperlinks both within and without, the terrain of the Web was re-visualized in terms of geography with Web sites. The notion of the Web portal was based on the idea that a wise provider would open a view of the Web that would protect us from chaos and provide uniform representations of information, but which might well lob us into sites and pages with diverse characteristics and an unexpected entrance into the Wild Wild Web, or else, if we wanted to stay safe, we could choose to remain penned up in walled gardens.

Names also change as technology and design advance. In the 1940s, for instance, some people had "car phones" that worked with radio technology. With the development of a reliable cellular system, we had handheld mobile devices that we called "cell phones." When mobile phones began to have something like a browser, some "apps," and messaging capabilities, we began to call them "smart phones." These phone names are not so much driven by metaphor as by somewhat naïve understandings of technology. But metaphors are still with us; for example, we see terms like "notebook" and "tablet" used to describe computers with certain dimensions and capabilities, even though one cannot typically sketch on, scribble on, or tear out and wad up a page.

A behavioral metaphor that has been rather more successful and pervasive is the notion of direct manipulation (discussed in more depth ahead), in which users can move objects about the screen in much the same way as they might in the physical world. Although the operations and conventions implicit in direct manipulation interfaces require more procedural learning than actually picking something up, the value of the metaphor is strong enough to boost most people up the learning curve.

Although interface metaphors can fail in many ways (as discussed later in this book), their prevalence has expanded the domain of interface design to admit contributions from specialists in graphic and industrial design, linguistics, psychology, education, and other disciplines. The metaphorical approach contributed to making interface design an interdisciplinary concern. They became lightning rods for people from many disciplines, either in service of or in reaction against them.

What is a User?

I resist using the word "user" in most contexts because it implies things we may not intend (drug users come to mind). In the context of human-computer interactions, "user" implies a power relationship and a kind of experience that tends to mischaracterize both technology and people. When we began to define human-computer interaction back in the 1970s and 1980s, the term "user" became quickly over-generalized. A person isn't typically defined as a "user" of the *New York Times* (unless you are house-breaking the dog) or of an automobile or a doctor. Over the years, I have exhorted my students not to use "user" unless it's really the correct word. For example, the "user" of a computer game is better characterized as a "player"; the "user" of an e-book is a "reader." Char Davies has called participants in VR experiences "immersants." Because this book covers a wide variety of human-computer interactions, I have used the word "interactor" as a general term, although I use "user" when I really mean it!

Interface Interdisciplines

While often driven by hardware innovation, the growing interdisciplinarity of interface design is also a product of heightened sensitivity to the experience of human-computer interaction. Change has been sparked by technology, scholarship, and imagination. The sections ahead are not complete histories; they contain brief sketches of exemplars and some comparisons to theatrical design.

IN THE BEGINNING, THERE WERE ENGINEERS Engineers were the first human-computer interface designers. Along that road, Douglas Engelbart and his team at Stanford Research International (SRI) were at the confluence of engineering, ease of use (human factors), and psychology and values, led by Engelbart's unwavering commitment to making the world a better place. Influenced early on by Vannevar Bush's canonical paper "As We May Think" (Bush 1945), Engelbart created a program at SRI called The Augmentation Research Institute (see Engelbart 1962). Its most famous invention was the computer mouse, but history often forgets that the group also invented hypertext, networked computers, and some of the foundations for graphical user interfaces, among other achievements.

What is a Computer?

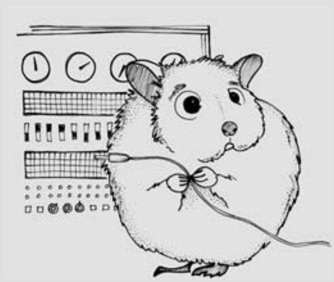
Computers were originally people. "Computer" was a title that described those people—mostly women—whose job it was to "perform the repetitive calculations required to compute such things as navigational tables, tide charts, and planetary positions for astronomical almanacs" (Kopplin 2010).

When I first wrote this book, the Macintosh computer was entering its fifth year in the marketplace. The personal computer was still a revolutionary device. Laptops, first envisioned by Alan Kay as what he called the "Dynabook" (Kay 1972), were developed. The Osborne Computer was designed in 1979 by Lee Felsenstein for Adam Osborne's company. Lee's primary design criterion was that the Osborne 1 had to fit under an airplane seat, and it did. The Grid Compass—the first successful "clamshell" portable—was released in 1981.*

Since those days, we have all seen an explosion of personal computing devices, from laptops to tablets to smartphones and smart wrist-watches, and change will keep coming. When I use the word "computer," I am speaking of it in the way it is defined by the *Oxford English Dictionary* (2013):

> *noun*
>
> *an electronic device which is capable of receiving information (data) in a particular form and of performing a sequence of operations in accordance with a predetermined but variable set of procedural instructions (program) to produce a result in the form of information or signals.*

I am using the term specifically in the domain of personal computing. The OED defines a personal computer as one that is designed for use by one person at a time. But if you consider a networked application such as a massively multiplayer online game, that definition may be misleading, since much of the processing is done on mainframe computers that are handling interactions from many people simultaneously, all in relation to one another. Where the code lives and which device is doing the bulk of the processing is not particularly relevant in my use of the term "computer." Most relevant to my argument are the representation produced and the interface affordances that shape and constrain human interaction with personal computers. We will ignore the tiny hamsters that run them.



* A somewhat less successful clamshell was introduced in Australia only about four months before the Grid Compass launched.

Engelbart believed that there were fantastic new horizons for human potential with computers. The notion of augmentation, while not a metaphor, was a vision that drove all of his work and solidified his team. It was only much later that the world rewarded him for it. He was ahead of his time, and the fledgling industry predictably pulled away key members of his team for shorter-term profit-making ventures.

Engelbart's legendary demo in 1968 was an incredible theatrical triumph as well as a technological one. Later nicknamed "the Mother of All Demos," Englebart sat on stage in San Francisco while his team was in Menlo Park. Engelbart recalls:

> Our computer was down at SRI in Menlo Park. In order to demo it, we beamed two channels of video along two microwave links up to San Francisco, bouncing them off dishes above the airport. There was only one video projector on the West Coast powerful enough for the conference hall, a Swedish Eidophor that I had to borrow from NASA. It was huge, maybe 6 feet tall. Then we rigged up a homemade modem—2,400 baud—to get signals from my console in San Francisco back to SRI over a leased line.
>
> On stage right was a big screen, 22 feet high. At the side of my display monitor, a camera pointed right at my face. Another camera was pointing down to capture my hands at the keyboard. It was pretty elaborate. My face would be on one side of the screen, with text on the other—or on a split screen with people in Menlo Park showing something as I talked about it. I'm told that this is the original videoconferencing demo (Jordan and Englebart 2004).

The theatre of the live performance and the skin-of-your-teeth presentation technology may sound familiar to theatre folk, but they were so far away from the culture of computing at the time that they made an indelible impact on the audience. Engelbart's demo lived on in the culture of SRI as well as the culture of PARC. The MIT Architecture Machine Group (later to become the MIT Media Lab) was still relying on demos that were often mock-ups, deconstructed as soon as they had been shown. At the same time, the content of the demo marked a major turning point in the practice, technology, and purpose of interface design.

ENTER THE PSYCHOLOGISTS Psychologists have been involved in the quest to understand human-computer interaction since the beginning of computing, through such disciplines as human factors design.[4] In the decade of the 1970s and on through the 1980s, cognitive psychologists developed critical and theoretical perspectives on human-computer interaction that were more focused on interface design than those of their colleagues in other branches of psychology. The work of Donald A. Norman, founder of the Institute for Cognitive Psychology at the University of California at San Diego, is especially illuminating. In the 1980s, Norman built a lab at UCSD that fostered some of the most innovative and germane thinking about human-computer interaction to that date (see Norman and Draper 1986 for a collection of essays by members and associates of this group). Norman's perspective is highly task-oriented. In his book *The Psychology of Everyday Things* (1988), Norman drives home the point that the design of an effective interface—whether for a computer or a doorknob—must begin with an analysis of what a person is trying to *do*, rather than with a metaphor or a notion of what the screen should display.

Norman's emphasis on action as the stuff that interfaces both enable and represent bores a tunnel out of the labyrinth of metaphor and brings us back out into the light, where what is going on is larger, more complex, and more fundamental than the way that the human and the computer "talk" to each other about it.

Norman's insights dovetail nicely with those of the "common ground" linguists, suggesting a notion of the interface that's more than screen-deep. The interface becomes the arena for the performance of some intentional activity in which both human and computer have a role. What is represented in the interface is not only the task's environment and tools, but also the process of interaction—the contributions made by both parties and the evidence of the task's evolution. I believe that Norman's analysis supports the view that interface design should concern itself with representing whole actions with multiple agents. This is, by the way, precisely the definition of theatre.

Norman was also a key figure in the development of another pivotal interface concept, the idea of direct manipulation. Direct manipulation

4. The literature in human factors and other psychological perspectives on human-computer interaction is huge. It is beyond the scope and purpose of this book to provide even a cursory survey of the entire domain. The work mentioned in this chapter is selected in terms of its relevance to the thesis of this particular book. Interested readers may wish to review *The Human Factor* by Richard Rubinstein and Harry Hersh, which includes an excellent bibliography; *Readings in Human-Computer Interaction*, by Ronald M. Baecker and Willam A.S. Buxton; or the various proceedings of ACM SIGCHI and the Human Factors Society.

interfaces employ a psychologist's knowledge of how people relate to objects in the real world in the belief that people can carry that knowledge across to the manipulation of virtual[5] objects that represent computational entities and processes. The term "direct manipulation" was coined by Ben Shneiderman of the University of Maryland, who listed these key criteria:

1. Continuous representation of the object of interest.
2. Physical actions or labeled button presses instead of complex syntax.
3. Rapid incremental reversible operations whose impact on the object of interest is immediately visible (Shneiderman 1982).

Shneiderman (1982) reported that direct-manipulation interfaces can "generate a glowing enthusiasm among users that is in marked contrast with the more common reaction of grudging acceptance or outright hostility." In a cognitive analysis of how direct manipulation works, Edwin Hutchins, James Hollan, and Don Norman suggest that direct manipulation as defined may provide only a partial explanation of such positive feelings. They posit a companion effect, labeled direct engagement: A feeling that occurs "when a user experiences direct interaction with the objects in a domain" (Hutchins et al. 1986). They add the requirements that input expressions must be able to make use of previous output expressions, that the system must create the illusion of instantaneous response (except where inappropriate to the domain), and that the interface must be unobtrusive.

It seems likely that direct manipulation and direct engagement are head and tail of the same coin (or two handfuls of the same elephant): one focusing on the qualities of action and the other focusing on subjective response. The basic issue is what is required to produce the feeling of taking action within a representational world, stripped of the "meta-context" of the interface as a discrete concern. Hutchins et al. sum it up this way: "Although we believe this feeling of direct engagement to be of critical importance, in fact, we know little about the actual requirements for producing it" (Hutchins et al. 1986).

5. The adjective "virtual" describes things—worlds, phenomena, etc.—that look and feel like reality, but which lack the traditional physical substance. A virtual object, for instance, may be one that has no real-world equivalent, but the persuasiveness of its representation allows us to respond to it as if it were real.

Nearly 20 years later, in his book *Emotional Design*, Norman (2004) says:

> We cognitive scientists now understand that emotion is a necessary part of life, affecting how you feel, how you behave, and how you think. Indeed, emotion makes you smart.

In more recent years, Norman might say that direct engagement arises from the emotional pleasure of a well-designed affordance; the characteristics of immediacy and lack of fussy procedural steps simply make direct manipulation feel good to us.

Here, I think, is an important articulation between psychology, interface design, and theatre. Direct engagement in the theatre arises first of all from real-time enactment and the enhanced attention it evokes. Audiences (and actors) have immediate emotional responses to the action on stage. Over the course of a play, emotions take on greater resonance, ideally producing empathy (literally, "feeling with" the characters). The interface (the venue, stage machinery, etc.) is not a matter of direct concern; when an audience is directly engaged with the action of the play, these elements literally disappear from conscious awareness. Further, theatrical audiences have an expectation of emotional pleasure. We will examine the nature of that pleasure in the next chapter.

Psychology is a familiar domain to dramatists, actors, and other theatre artists because of its focus on the human mind, behavior, and emotions. Understanding how psychology and theatre are alike and different may illuminate the distinct contributions that each can make in the field of human-computer interaction. The two domains have several elements in common. Both concern themselves with how agents relate to one another in the process of communicating, fighting, solving problems, building things, having fun—the whole range of human activity. Both interpret human behavior in terms of emotions, goals, conflicts, discoveries, changes of mind, successes, and failures. Both observe and analyze human behavior, but each employs those means to different ends: In general, psychology attempts to understand what goes on with humans in the real world with all their fuzziness and loose ends, while theatre means to represent a kind of thing that might go on, simplified for the purposes of logical and affective clarity. Psychology explicates human behavior, while theatre represents it in a form that provides intellectual and emotional closure. Theatre is informed by psychology, but it turns a trick that is outside of psychology's province through the direct representation of action.

GRAPHIC DESIGN, ANIMATION, AND SOUND In the analyses of both Shneiderman (1987) and Hutchins et al. (1986), continuous representation and physical action depend heavily upon graphical representation. Hutchins identifies the granddaddy of direct manipulation as Ivan Sutherland's graphical design program *Sketchpad* (Sutherland 1963). Graphical (and, by extension, multisensory) representations are fundamental to both the physical and emotional aspects of directness in interaction.

In many ways, the role of the graphic designer in interface design is parallel to the role of a theatrical scene designer. Both create representations of objects and environments that provide a context for action. In the theatre, the scene designer provides objects like teacups and chairs (props), canvas-covered wooden frames that are painted to look like walls (flats), and decorative things like draperies and rugs (set dressing). The behaviors of these elements are also designed: doors open, make-believe bombs explode, trick chairs break in barroom brawls. The lighting designer uses elements of color, intensity, and direction to illuminate the action and its environment and to focus our attention on key areas and events. In interface design, animation has been used increasingly as processing power has grown.

Scene and light designers use such elements as line, shadow, color, texture, and style to suggest such contextual information as place, historical period, time of day, season, mood, and atmosphere. Theatrical designers also employ metaphor (and amplify the metaphors provided by the playwright) in the design of both realistic and non-realistic pieces: The looming cityscape around Willy Loman's house in *Death of a Salesman* metaphorically represents his isolation and the death of his dreams; abstract webs of gauzy fabric suggest the multiple layers of illusion in the personality of *Peer Gynt*. At Ohio State University, the Advanced Computing Center for the Arts and Design (ACCAD) has collaborated with the Departments of Theatre and Dance to produce real-time visual effects, including characters projected from the motion-capture studio onto the stage where they interact with "live" actors.

In interface design, graphic designers and animators make the same sorts of contributions. They render the objects and environments in which the action of the application or system will occur, imparting behaviors to some objects and representing both concrete and ephemeral aspects of context through the use of such elements as line, shadow, color, intensity, texture, and style. Such familiar metaphors as desktops and windows provide behavioral and contextual cues about the nature of the activity that they support.

Sounds Good to Me

The music in some of the early arcade and box games was truly horrible. Tim Vasikalis, an extremely successful composer and producer of sound for games, explains that "early video game music was exclusively developed by the engineers themselves. . . . Back in the day, the only way to embed sound into a game was by directly programming it into the computer chips" (Vasikalis 2012). Oh, that explains it.

Despite *Pac Man*, it became clear to me in the early 1980s that audio had tremendous potential in computer games. This belief began with the simple observation that when the radio is on, my brain visualizes the action and characters, but if the TV is on without sound, my brain does not create a soundtrack. Audio-only videogames generally met with curiosity, but not great success in the general marketplace (e.g., *Real Sound* by Kenji Eno, 1997) until spatialized audio showed up. Binaural recording has enabled spatialized sound for VR as well as for absorbing audio games (e.g., the mobile game *The Nightjar*).

I first used binaural field recording to create spatialized sound in the production of *Placeholder*, a virtual reality project at the Banff Centre for the Arts in 1993. One of the soundscapes we needed was the sound of a waterfall, including walking through it. I asked my partner Rob to do the recording, which involved wearing special microphones on either side of his head. As it was a waterfall, we had to do something to protect the microphones. A search party was sent to town to get some condoms, which were then placed over the mics. So here was this guy, struggling through a rushing creek with strange affordances on his head, then standing (briefly) under falling water that was hitting his head at about 300 psi. At that moment I wondered what a random hiker would have thought.

Two big things came out of the *Placeholder* recordings. One was that, although the piece represented three scenes using very different videographic methods, the audio held the world together. The other was that Rob and I have continued to make environmental binaural recordings since then, and it's been a blast.

Sound and music design in interactive media—especially games—has become increasingly important and sophisticated. The introduction of spatialized sound into computer games in the late 1990s accelerated the development of sound-design tools, technology, expertise, and curricula. Simpler sounds give us cues as well as a sense that something "real" is going on, from the minimalist "whoosh" when you send a message to the "crackle" of "paper" when you drag something to the "trash" on the Mac.

Both theatrical design and interface design are aimed at creating repre-sentations of worlds that are like reality, only different. But a scene design is not a whole play; for that we also need representations of character and action. Likewise, the elements of interface design are only part of the whole representation that we call human-computer interaction. How should an interface come to be? In effective interaction design, the interface does not come last; it develops throughout with the entire design process. It is deeply entwined with functionality. It shows sensitiv-ity to the interactor and sometimes even constrains functionality that can-not or need not be touched effectively by the interactor. If we think of an application as an organic whole, the process by which it is created should be organic as well.